Quantitative Assessment of Robot-generated Maps

C. Scrapper, R. Madhavan, R. Lakaemper, A. Censi, A. Godil, A. Wagan, and A. Jacoff

Abstract Mobile robotic mapping is now considered to be a sufficiently mature field with demonstrated successes in various domains. While there has been much progress made in the development of computationally efficient and consistent mapping schemes, it is still murky at best on how these maps can be evaluated. We are motivated by the absence of an accepted standard for quantitatively measuring the performance of robotic mapping systems against user-defined requirements. It is our belief that the development of standardized methods for quantitatively evaluating existing robotic technologies will improve the utility of mobile robots in already established application areas, such as vacuum cleaning, robot surveillance, and bomb disposal, but will also enable the proliferation and acceptance of such technologies in other emerging markets. This Chapter summarizes our preliminary efforts by bringing together the research community towards addressing this important problem which has ramifications not only from a research perspective but also from consumers', robot manufacturers', and developers' viewpoints.

C. Scrapper, R. Madhavan, A. Jacoff
Intelligent Systems Division, National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899, USA, e-mail: chris.scrapper@nist.gov, raj.madhavan@nist.gov, adam.jacoff@nist.gov

R. Lakaemper
Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, USA, e-mail: lakamper@temple.edu

A. Censi
Control & Dynamical Systems, California Institute of Technology, Pasadena, CA 91125, USA, e-mail: andrea@cds.caltech.edu

A. Godil, A. Wagan
Information Technology Laboratory, National Institute of Standards and Technology (NIST), Gaithersburg, MD 20899, USA, e-mail: afzal.godil@nist.gov, asim.wagan@nist.gov

1 Introduction

Mobile robots permitting collaborative operations of man and machine present a new frontier of research with almost limitless possibilities by serving as an indispensable aid in difficult, and unstructured environments. Robots will play an increasingly vital role in assisting humans in a variety of domains ranging from innocuous daily chores around the household to potentially harmful situations. The use of robots, either tele-operated or autonomous, in dangerous situations can not only save lives but also can improve productivity (e.g. factory floors) and in some cases provide solutions which are not possible by humans alone (e.g urban search and rescue). It is not hard to see that the ability to build a map of the working environment is a desirable feature in many domains of interest. For example, in a disaster scenario concerning extrication of victims, a robot generated map will serve as in invaluable tool for the first responders.

Not surprisingly, the development of efficient robotic mapping algorithms have received their due attention from roboticists. A myriad of approaches have been proposed and implemented, some with greater success than others. The capabilities and limitations of these approaches vary significantly depending not only on the operational domain, and onboard sensor suite limitations, but also on the requirements of the end user: Will a 2D map suffice as an approximation of a 3D environment? Is a metric map really needed or is it enough to have a topological representation for the intended tasks or do we need a hybrid metric-topological map [36]? It is thus essential for both the developers and the consumers (probably to a lesser extent) of robotic systems to understand the performance characteristics of employed methodologies which will allow them to make an informed decision.

To the authors' knowledge, there is no accepted standard for quantitatively measuring the performance of robotic mapping systems against user-defined requirements; and furthermore, there is no consensus on what objective evaluation procedures need to be followed to deduce the performance of these systems. For instance, currently, the evaluation of robotic maps is based on qualitative analysis (i.e. visual inspection). This approach does not allow for better understanding of what errors specific systems are prone to and what systems meet the needs. It has become common practice in the literature to compare newly developed mapping algorithms with former methods by presenting images of generated maps. This procedure turns out to be suboptimal, particularly when applied to large-scale maps.

The lack of reproducible and repeatable test methods have precluded researchers working towards a common goal from exchanging and communicating results, intercomparing robot performance, and leveraging previous work that could otherwise avoid duplication and expedite technology transfer. This lack of cohesion in the community hinders the progress in many domains, such as manufacturing, service, search, rescue, and security. Providing the research community access to standardized tools, reference data sets, and an open-source library of solutions, researchers and consumers will be able to evaluate the cost and benefits associated with available technologies. The development of standardized methods for quantitatively evaluating existing robotic technologies will not only improve the utility of mobile robots in already established application areas, such as vacuum cleaning, robot surveillance, and bomb disposal, but will enable the proliferation and acceptance of such technologies in other emerging markets.

Some researchers have recognized the need for quantitative evaluation of mapping and position estimation schemes and are attempting to address it through several programs. For example, the Robotics Data Set Repository (Radish) provides a collection of standard robotics data sets [3]. The OpenSLAM repository contains collections of source codes of various SLAM algorithms [2]. While a step in the right direction, they do not address objective performance evaluation and replication of algorithms is not straightforward. Emerging standard test methods for emergency response robots, developed by the National Institute of Standards and Technology (NIST) and the Department of Homeland Security (DHS), have been developed impart to provide the research community with an efficient way to test their algorithms. These test methods can be proliferated widely to minimize the costs associated with maintaining functional robots and traveling to one of the permanent arena sites for validation and practice.

In [29], map quality is assessed using conditional random fields. The assessment is proposed as an introspective inspection of workspace representations towards analyzing the reliability/plausibilty of the representation. A single 3D laser map is segmented into planar patches based on neighboring relations into 'plausible' and 'suspicious' using a context-sensitive classification framework. The proposed framework can be thought of as a qualitative assessment based on quantitative metrics. It is not clear how to extend this method to assess and compare quality of two maps of the same area.

Many researchers have suggested using vision rather than laser rangefinders for mapping purposes [23, 35, 41]. Though a passive sensor, cameras are an attractive option due to their low consumption of power, relatively low cost, and ability to provide large bandwidth of information. In [35], the authors have a developed testbed infrastructure as a vision SLAM benchmark using synchronized inertial measurement unit, GPS and stereo images in an outdoor setting. The proposed benchmark may provide a mechanism for comparing two maps generated by using the data collected via the proposed infrastructure and geographically referenced aerial images as ground truth. While feasible in some cases, the assumption of ground truth is, in general, an overly restrictive assumption.

The RoboCup Rescue competitions have proved to be a good forum to evaluate task-based performance of robots. An image similarity metric and a cross entropy metric are outlined in [21] to measure the quality of occupancy grid maps. The metric gives an indication of distortion of the map with respect to a ground truth map in the presence of noise and pose errors. This metric is embedded in the Jacobs Map Analysis Toolkit [1] and has been tested for comparing maps in the RoboCup context.

While contributions by individual researchers are important steps in the right direction to overcome technological barriers to robotic mapping, a concerted effort among all interested parties is crucial. The primary focus of our efforts is to thus bring together researchers, consumers, and vendors to define objective methodologies for quantitatively evaluating robot-generated maps [34, 8].

Standard test methods are vital in establishing a confident connection regarding the expectations and performance objectives of robotic technologies between developers and consumers. They consist of well-defined testing apparatuses, procedures, and objective evaluation methodologies that isolate particular aspects of a system in known testing conditions [4]. This provides developers with a basis for understanding the objective performance of a system and allows consumers to confidently select systems that will meet their requirements.

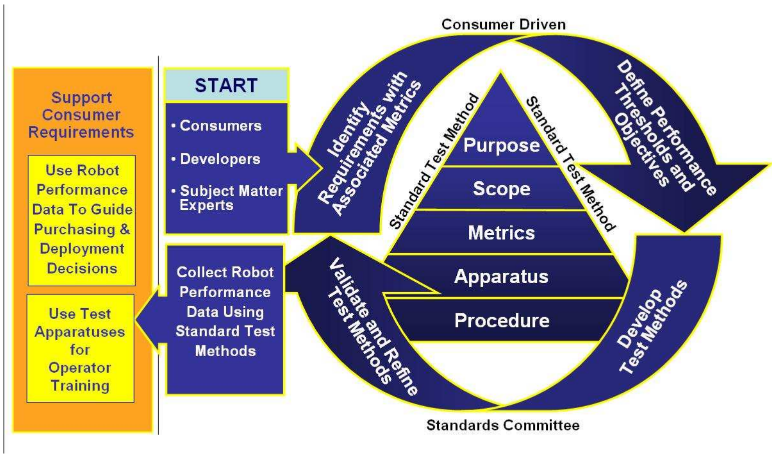

Fig. 1 The standard test method developmental cycle used by NIST and DHS.

In order to ensure the integrity of the test methods, it is essential to use a developmental cycle, shown in Figure 1, that continuously reassesses the validity of the test methods. This process starts with a comprehensive analysis of the application domain to identify requirements with associated metrics, thresholds of performance, and best-case performance objectives. This analysis provides the basis for developing test methods, procedures, and testing scenarios that are intentionally abstract so as to be repeatable across a statistically significant set of trials and reproducible by other interest parties.

The remainder of this chapter is dedicated to the development of standard test methods and techiques for evaluating robotic mapping using the cycle described above. Section 2 provides a brief description of the emerging test apparatuses used to challenge robotic mapping in specific ways. Section 3 develops a theoretical approach, using the Cramer-Rao Bound (CRB), to assess the objective performance and thresholds of pose and mapping estimates. The subsequent Sections 4 and 5 outline two experimental techniques used to quantitatively assess the quality of robot-generated maps. Section 6 provides conclusions and continuing research.

2 Developing Test Scenarios for Robotic Mapping

The performance of any given robotic mapping system is largely dependent on its ability to reliably accomplish two fundamental tasks. First is the ability of the system to make accurate measurements of its surrounding environment. Second is the ability of the system to reliably determine valid correspondences between observations, i.e. associating an object in one observation with its counterpart in another. The type of environment and the conditions found in that environment strongly influence the ability of the system to accomplish either task. Furthermore, subtle differences in relatively similar environments may have very different effects on the overall performance of the system.

As noted eariler in Section 1, the evaluation of robot-generated maps is often based on a qualitative approach that does not take into account how specific environmental conditions impact the performance of the system. While this type of analysis provides some indication of the overall performance, it does not allow researchers to understand what errors a specific system is prone to, how these errors impact the overall performance of that system, and how performance of that system compares with competing approaches. When developing standard test methods for evaluating robot-generated maps, it is important to develop repeatable and reproducible testing scenarios that isolate potential failure conditions in a controlled environment.

The remainder of this section summarizes a suite of test apparatuses designed to classify the performance of robotic mapping systems over a range of application requirments. With each test apparatus focused on challenging the system with varying levels of environmental complexities, this suite is intended to provide a comprehensive evaluation that will serve as the baseline for comparison and will help developers refine the capabilities of their system (or address limitations). Section 2.1 provides a brief discription of the process used to develop the test apparatuses in this suite. Sections 2.1.1, 2.1.2, and 2.1.3, introduces the resulting prototype apparatuses currenly used to characterize mapping systems.

2.1 Performance Singularity Identification and Testing

Identifying *performance singularities*, or the point where the mapping system fails to be well-behaved, is essential for understanding the impact of the environment on the overall performance of the mapping system and what environmental conditions are problematic for robotic mapping systems in general. *Performance singularity identification and testing* [37, 38] defines a two-pronged approach by which one can systematically evaluate the impact of environmental conditions(that may contribute to the occurrence of performance singularities) and analyze the impact of these singularities on the overall performance of the system.

The first step in this approach is to evaluate the performance at the system level to identify divergences in the performance of the mapping system. Using ground truth information about the location of the robot and the surrounding environment, the *performance evaluation* step facilitates the decomposition of the errors arising in the pose estimate. This enables the discover of irregularities and helps identify environmental situations where a performance singularity has occurred.

The second step in this approach is to analyze performance of a mapping system at the algorithmic level to gain insight into the cause and repercussions of the performance singularity previously identified. *Performance analysis* takes advantage of the ground truth to measure the error in the pose estimate at each discrete observations. This produces a convergence profile elucidates the convergence characteristics, such as the stability of the pose estimate.

2.1.1 The Maze: Scenarios with Distinct Features

Fig. 2 The Maze is a testing scenario that limits complexities in the environment to evaluate the objective performance of robotic mapping systems. The top images show the overall size to be 10 meters by 15 meters. The bottom left image shows the continuous 15°pitch and roll ramp flooring foudn throughout the maze. The bottom right image shows additional mapping features, such as concave and convex surfaces.

The Random Maze apparatus has distinct features, which provide the best-case scenario, where mapping systems should perform optimally. As seen in Figure 2, this apparatus contains a closed set of distinct mapping features and vertical walls that produces unique observations. This enables mapping systems to associate features, increasing the likelihood of determining valid correspondences. Perpendicular surfaces, which allow for more accurate measurements of the surrounding environment, appear in almost every scan. Limiting the environmental complexities allows developers to tune their systems and establishes baseline a for comparison. The modularity of the apparatus enables the randomization of the maze configurations for repetitive testing. Common building materials makes it a low-cost and easy to replicate.

2.1.2 The Tube Maze: Scenarios with Occluded Features

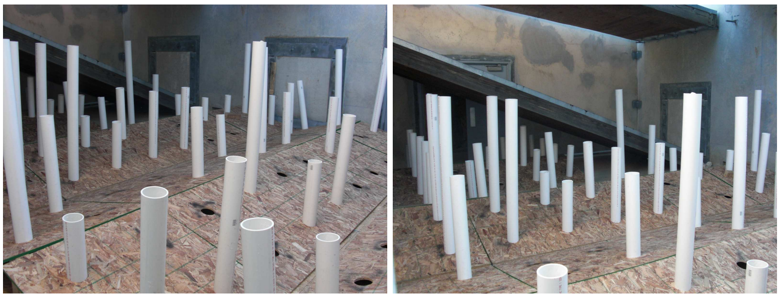

Fig. 3 The Tube Maze, which also has continuous 15°pitch and roll ramp flooring, is designed to test the ability of the system to reliably determine valid correspondences.

The Tube Maze apparatus challenges the mapping systems ability to determine valid correspondences. As seen in Figure 3, this apparatus also contains a closed set of distinct features where nearby features periodically occlude more distant features as the robot moves through the environment. This produces a situation where consecutive observations may not contain the same set of features, increasing the likelihood of correspondence errors. However, the nearby features, not occluded, enable the system to make accurate measurements of the immediate vacinity, helping the system avoid catastrophic failures. This occurs frequently in unstructured environments and is a key component to a successful mapping system. Similarly, the modularity of the apparatus and common building materials provide a simple way to validate successful mapping.

2.1.3 The Tunnel: Scenarios with Minimal Features

The Featureless Tunnel apparatus implements the degenerative case for mapping algorithms. As shown in Figure 4, this apparatus presents a symmetric and featureless environment to the mapping system, inhibiting the system's ability to make accurate measurements of its environments and determine valid correspondences. The only distinct feature in the apparatus is the turn where the far wall is perpendicular to the robot. The lack of distinct features increases the potential for catastrophic errors by preventing the convergence of the pose estimate in the mapping system. While this

Fig. 4 The Featureless Tunnel apparatus challenges the aspects of robotic mapping systems that require correspondences to estimate the robot's pose. The apparatus is 15 meters with a single turn, 15°roll ramp flooring, and black felt covered interior walls.

situation does not occur commonly (except in culverts, sewers, and tunnels), this testing apparatus is essential to understanding how the system fails.

3 Assessing Objective Performance Using Theoritical Analysis

The test methods as developed provide a basis for evaluation of a range of application requirements with associated metrics, thresholds of performance, and best-case performance objectives. Theoretical analysis of mapping systems assists can provide insight into the range of performance in application domains and how these systems can be improved.

We use the following notation. Let $x_k \in \mathrm{SE}(2)$ be the robot pose at time k; let $\delta_k \in \mathrm{SE}(2)$ be the incremental motion of the robot, such that $x_{k+1} = x_k \oplus \delta_k$, where $\oplus$ is the pose composition operator (group operation on SE(2)). Let w represent the "world" or "map". Let y_k be the sensor readings. The sensor model can be specified either in a functional form such as $y_k = f(x_k, w) + \varepsilon_k$, where ε_k is a noise term, or alternatively using the distribution $p(y_k|x_k, w)$. With this notation, we formalize *localization* as the problem of estimating x_k given the observations $y_{1:k}$ and a *known* map w. We call *pose-tracking* the problem of recovering the robot displacements δ_k from the sensor readings, without knowing the map. Pose tracking using range scans is commonly called *scan matching*. We call *mapping* the problem of recovering w

given known poses. Finally, SLAM is the problem of recovering both poses and map at the same time. This classification is useful in this particular analysis; but, in practice, distinctions are blurred between these problems. For example, some algorithms such as ICP are applicable to both localization and pose tracking, and a complete solution may decompose the SLAM problem in smaller subproblems involving pose-tracking, mapping, and (re)localization.

Industrial deployment of localization/SLAM algorithms implies matching the algorithm properties to the operational requirements. *Computational properties,* such as speed and memory consumption, are easy to define and measure directly. *Statistical properties* make sense for most algorithms that work in a probabilistic formulation of the problem. Such properties are the accuracy (estimate covariance), the presence of a bias, the consistency (whether the algorithm has a good estimate of its actual accuracy). These properties are easy to define mathematically, but might be hard to measure in practice because a ground truth is needed. Equally important are the *robustness properties*, which refer to the ability for the algorithm to work even if the assumptions on which it relies are slightly violated. For example, a localization algorithm should not fail completely if the provided map is only slightly different than the actual environment. Likewise, it should not fail if the sensor has a covariance slightly larger than the assumed one. The robustness properties are hard to define analytically, because by definition they refer to unknown violations of the assumptions. Finally, the output of some algorithms, such as environment maps, might be used by both machines and humans, but this *"user-friendliness"* cannot be be defined mathematically.

All these desirable properties are sometimes contrasting. For example, speed versus accuracy is an obvious trade-off in many algorithms. Other typical trade-offs include robustness versus accuracy (for example, this appears in choosing the percentage of measurements to discard as outliers) and generality versus accuracy (an algorithm which makes more assumptions about the environment can be more precise than one that works in more situations).

There are essentially two ways to verify these properties: either using benchmarks, or using theoretical analysis. These two are complementary activities: a theoretical analysis can be done only on some kind of idealized model of the system and the algorithm; the benchmarks can verify whether the assumptions are verified, for the actual implementation in the actual environment. Conversely, benchmarks are incomplete without analysis, because they do not explain *why* the algorithm behaves in a certain way, and *whether* and *how* the algorithm can be improved.

3.1 The case for statistical bounds

In practice, it might be unfeasible to prove analytically that a particular localization/SLAM algorithm has one of the above mentioned properties. In estimation theory, there are a number of ready-made results for the canonical estimators, such as the maximum-likelihood estimator, regarding their accuracy and consistency. However, this kind of results are not easily transferred to the actual algorithms, because of the ad-hoc approximations that are necessary in the implementations.

For example, in the Bayesian framework, the solution to the the filtering problem is given by a recursive formula which has a closed form. Most SLAM papers start with this uncomputable formula, and start simplifying it using various assumptions, until an approximation which can be computed efficiently is obtained. The problem is that, once the symbol "$\simeq$" is used, one loses any guarantee about the properties of the resulting algorithm. Another canonical example of theoretically sound, but hard to analyze, estimators are particle filters. The only strong results regard the asymptotic behavior as the number of particles goes to infinity, but nothing is guaranteed for a finite computation [11]. The answer to the question of how many particles does one actually need is usually very fuzzy [12]. These problems are specific to "dense" algorithms. In the "discrete" case, for Extended-Kalman Filter-based methods, it is easier to do theoretical analysis because the complexity of actual sensors has been abstracted away into bearing/range observations of landmarks: hence we know that the EKF is inconsistent, where the inconsistency originates, and what to do about it [13, 20, 18].

Because an explicit white-box analysis of a localization/SLAM algorithm might be unfeasible, a possible first step is to investigate the problem itself, for example by considering the statistical bounds to the problem.

The theory of statistical bounds is well developed and offers many tools [40]. The Cramr-Rao Bound (CRB) is a classic one which is easy to derive and to use. In the nonlinear case with additive gaussian noise ($y = \mathbf{f}(x) + \varepsilon$), one defines the Fisher Information Matrix (FIM) as $\mathscr{I}[x] = \frac{\partial \mathbf{f}}{\partial x}^{\mathrm{T}} \Sigma^{-1} \frac{\partial \mathbf{f}}{\partial x}$, with Σ being the covariance of the noise. Then the CRB establishes that, for any unbiased estimator, $\mathrm{cov}[\hat{x}] \geq (\mathscr{I}[x])^{-1}$; if the estimator is biased, $\mathrm{cov}[\hat{x}] \geq \left[I + \frac{d}{dx}\mathrm{b}_{\hat{x}}(x)\right] (\mathscr{I}[x])^{-1} \left[I + \frac{d}{dx}\mathrm{b}_{\hat{x}}(x)\right]^{\mathrm{T}}$, where $\mathrm{b}_{\hat{x}}$ is the bias of the estimator. In general, the CRB is not tight, except is special cases, such as when $\mathbf{f}$ is linear; the CRB is approximately tight at high signal-to-noise ratios.

The CRB is an useful tool with many uses. It provides a lower bound for the accuracy that is a baseline for comparing the actual experimental results. It allows to verify the realism of accuracy claims, and the proper execution of experiments. After it has been proved to be tight, it can also be used to predict the actual covariance.

Nevertheless, it is worth pointing out some intrinsic limitations of this kind of analysis. This theory applies only when localization is modelled probabilistically, and it only models the effect of stochasticity in the readings, which is only one of the many sources of error in the algorithms (others are, e.g., convergence to local minima). Moreover, this theory only gives negative results; establishing positive results of guaranteed accuracy must still be done on a case-by-case basis.

3.2 The CRB for range-finder localization

We define localization as estimating the pose given a perfect map and a range scan. This case has been considered in the paper [9], from which we recall the main results. Assume the pose of the robot is $x = (t, \theta)$, and the output of the range finder is $\{\tilde{\rho}_i\}_{i=1}^{n}$ where $\tilde{\rho}_i$ is the i-th ray along direction φ_i. The FIM is a 3×3 symmetric semidefinite positive matrix which can be computed as follows:

$$\mathscr{I}[x] = \sum_{i=1}^{n} \frac{1}{\sigma_i^2 \cos^2 \beta_i} \begin{bmatrix} v(\alpha_i)v(\alpha_i)^{\mathrm{T}} r_i \sin(\beta_i)\, v(\alpha_i) \\ * \quad\quad r_i^2 \sin^2(\beta_i) \end{bmatrix} \tag{1}$$

In this expression, $v(\alpha_i)$ is the versor corresponding to the surface normal direction α_i, $\beta_i = \alpha_i - (\theta + \varphi_i)$ is the incidence angle, and r_i is the distance to the obstacle. The FIM depends both on the environment, and the particular pose of the robot in the environment: there are parts of the environment where localization is easier than in others.

By computing the CRB as $(\mathscr{I}[x])^{-1}$, one obtains the achievable accuracy in a particular environment and pose. Experiments show that the CRB is approximately strict; in localization, the high signal-to-noise condition corresponds to having the sensor standard deviation σ negligible with respect to the size of the environment, which is usually the case. Given the FIM for "one shot" localization, the covariance over a trajectory can be evaluated by propagating the CRB through the system dynamics, according to a standard procedure [40].

The FIM can be used also in a more qualitative way to study the observability of the problem: the FIM drops rank in under-constrained situations (corridor or circular environment).

Thus it is possible to do a fairly complete characterization of localization. The reason is that, being a finite dimensional problem, all is needed is a straightforward use of the basic tools of statistics. However, this cannot so easily extended to pose tracking.

3.3 The CRB for one-shot pose tracking

We now consider "one-shot" pose tracking, in which we estimate the robot displacement given two sensor readings. Let $x_1 \in \mathrm{SE}(2)$ be the first pose, $\delta \in \mathrm{SE}(2)$ be the robot motion, and therefore $x_2 = x_1 \oplus \delta \in \mathrm{SE}(2)$ be the second pose. The sensor model reads $y = \mathbf{f}(x, w) + \varepsilon$, with w now an unknown map. Because the map is unknown, pose-tracking is qualitatively different from localization, and conceptually closer to full SLAM. Moreover, the problem is ill-posed if a prior for the map is not specified. Because of the unknown map, using the CRB is inconvenient, as one should: 1) choose a particular (differentiable) parametrization of w; 2) define the prior for w; 3) use a variant of the CRB such as the Bayesian Cramer-Rao bound [40].

However, the paper [10] shows that there is a "trick" one can use to obtain accuracy bounds without considering the prior distribution of the map. The result is that a lower bound for the FIM of δ is:

$$\mathscr{I}[\delta] \leq \left((\mathscr{I}[x](x_1))^{-1} + (\mathscr{I}[x](x_2))^{-1} \right)^{-1} \tag{2}$$

In this expression, $\mathscr{I}[x](x_1)$ and $\mathscr{I}[x](x_2)$ is the FIM for localization of x, evaluated at the poses $x = x_1$ and $x = x_2$ respectively. This bound is significant because it depends neither on the representation, nor on the prior used for the map. Therefore, it allows to reduce the analysis of pose-tracking, an infinite-dimensional problem that involves both w and x, to the analysis of localization, a finite dimensional problem that involves only x.

The bound in (2) can be very optimistic, but it can also be shown that this is also the "best possible" bound, in the sense that there is always a certain prior for the map such that equation (2) holds with equality. Moreover, it can be shown that (5) holds with equality in the limit $\delta \to 0$. This means that, for small steps, the prior for the map is unimportant; the data itself is the model.

3.4 The CRB for pose tracking over a trajectory

Let us consider now the problem of evaluating the accuracy of pose tracking over a trajectory. It might be counterintuitive, but it is not necessarily true that the error of pose tracking grows linearly with the number of steps. The reason is that scans are matched pairwise: scan y_n is used for estimating both $\delta_{n-1} = x_n \ominus x_{n-1}$ and $\delta_n = x_{n+1} \ominus x_n$. The errors on δ_n and δ_{n-1} are now correlated, and because of this correlation, the covariance of the cumulative estimate $\delta_1 \oplus \cdots \oplus \delta_n$ is not just the sum of the uncertainties anymore. This effect must be taken into account when propagating the uncertainty [32], and it is likely to be an important effect to consider when deriving accuracy bounds [31].

As it turns out, for relative sensors this is actually a positive effect: errors tend to cancel out. A one-dimensional toy example can show this point. Suppose the robot is moving on a line; $x \in \mathbb{R}$, and there is a single wall at point $w \in \mathbb{R}$. The range-finder then measures a single reading $y = w - x$ which is the distance to the wall. Assume now that there are n steps going from pose x_1 to x_{n+1}, with $n+1$ range finders readings defined as: $y_i = (w - x_i) + \varepsilon_i$. Assuming we are doing pose tracking, we would estimate first the incremental displacement, by computing

$$\delta_i = y_{i+1} - y_i \tag{3}$$

Then, we would combine the incremental estimates as to obtain the cumulative estimate

$$\delta_{1:n} = \delta_1 + \delta_2 + \cdots + \delta_n \tag{4}$$

What is the covariance of $\delta_{1:n}$? A wrong answer would be the following. From (3), it is clear that $\text{cov}[\delta_i] = 2\sigma^2$ if σ^2 is the covariance of the single reading. The naive way would be to sum covariances and obtain $\text{cov}[\delta_{1:n}] = 2n\sigma^2$. But this is not correct because the δ_i are correlated. In fact, equation (4) can be rewritten as: $\delta_{1:n} = \delta_1 + \delta_2 + \cdots + \delta_n = (y_2 - y_1) + (y_3 - y_2) + \cdots + (y_{n+1} - y_n) = y_{n+1} - y_n$. Because the ys cancel out, the errors compensate exactly, and one then can conclude that $\text{cov}[\delta_{1:n}] = 2\sigma^2$, which is independent of the number of steps. This is an extreme example which corresponds to a trivial system. For nonlinear systems, the errors will not cancel perfectly. Still, this example shows that, in general, the bound over a trajectory is

$$\text{cov}(\delta_1 \oplus \cdots \oplus \delta_n) \geq (\mathscr{I}[x](x_1))^{-1} + (\mathscr{I}[x](x_{n+1}))^{-1} \tag{5}$$

This is, however, very optimistic for actual scan matching; but it is the best one can do without considering the map.

3.5 The CRB for mapping and SLAM

We turn now to the problem of mapping (estimating the map given the readings of known poses) and SLAM (where also the poses are unknown). Establishing accuracy bounds on the accuracy in estimating the map is more laborious.

The dominant phenomenon is that asking what is the achievable accuracy of the map is an ill-posed question if the prior of the map distributions is not specified. Consider the following toy example. Suppose that the world is allowed to have only two shapes: triangle ($\triangle$) and circle ($\bigcirc$), and that they are equally likely. Formally, we set the world set $\mathcal{W} = \{\triangle, \bigcirc\}$ and the prior $p(w = \bigcirc) = p(w = \triangle) = 0.5$. A decent sensor can distinguish exactly between the circle and the triangle. Therefore, the error of mapping is zero (or conversely, the accuracy is infinite): once one decides which one of the two shapes is correct, the reconstruction is perfect. The same holds even in the case where $\mathcal{W}$ is a much larger set, but $\triangle, \bigcirc$ are the only objects having non-zero prior. Therefore, the achievable accuracy depends arbitrarily on the prior.

Another example is the following. An "unstructured" environment has many more degrees of freedom than a "structured" one, which, for the most part, could be described even by a finite-dimensional representation. Therefore, it can be seen intuitively that, if the same sensor is used in both kinds of environments, the achievable accuracy in reconstructing the map is higher in the structured environment. However, to quantify this intuition, one needs to: 1) formally define the set $\mathcal{S}$ of structured environments; 2) formally define the set $\mathcal{U}$ of unstructured environments; 3) embed both in a larger set $\mathcal{W}$, for example the set of all closed curves; 4) restate the structured/unstructured hypotheses by defining appropriate structured/unstructured priors on $\mathcal{W}$; 5) apply one of the Bayesian bounds to derive that, yes, indeed, mapping is easier in a structured environment.

This formal reasoning about shapes has not been used in SLAM research yet, while it is used in other fields such as computer vision and stochastic geometry. An "intrinsic" theory of shape can be used to discuss the properties of shapes and shape distributions independently of their representation; see, e.g., the classical work [27] for points distributions and for curves [30, 24]. Such an approach is necessary for establishing meaningful bounds on the accuracy of mapping and SLAM.

4 Evaluating Local Metric Consistency of Robot-Generated Maps Using FFS and Virtual Scans

In this section, we discuss how the integration of low level spatial cognition processes (LLSC) and mid level spatial cognition processes (MLSC) can help to improve the performance of robot mapping, and how a LLSC/MLC system can be used for map evaluation.

In robot cognition, MLSC processes infer the presence of mid level features from low level data based on regional properties of the data. In our example case, we detect the presence of simple mid level objects, i.e. line segments and rectangles. The MLSC processes model world knowledge, or assumptions about the environment. The example assumes the presence of (collapsed) walls and other man made structures. If possible wall-like elements or elements resembling rectangular structures are detected, our system generates the most likely ideal model as a hypothesis, called 'Virtual Scan'. Virtual Scans are generated from the ideal, expected model in the same data format as the raw sensor data, hence Virtual Scans are added to the original scan data indistinguishably for the low level alignment process; the alignment is therefore performed on an augmented data set.

In robot cognition, LLSC processes usually describe feature extraction processes based on local properties like spatial proximity. An example is metric inference on data points (laser scanner reflection points). In our system laser scans (virtual or real) are aligned to a global map using mainly features of local proximity using the LLSC core process of 'Force Field Simulation' (FFS). FFS was recently introduced to robotics [26].

In FFS, each data point can be assigned a weight, or value of certainty. It also does not make a hard, but a soft decision about the data correspondences as a basis for the alignment. This is achieved by computation of a correspondence probability to multiple neighboring points, based on weight, distance and direction of underlying linear structures. Mainly these features makes FFS a natural choice over its main competitor, ICP [7, 33], for the combination with Virtual Scans (however, the general idea of Virtual Scans is applicable to both approaches). The weight parameter can be utilized to indicate the strength of hypotheses, represented by the weight of virtual data.

FFS is an iterative alignment algorithm. The two levels (LLSC: data alignment by FFS, MLSC: data augmentation) are connected by a feedback structure, which is repeated in each iteration:

- The FFS-low-level-instances pre-process the data. They find correspondences based on low level features. The low level processing builds a preliminary version of the global map, which assists the mid-level feature detection
- The mid level cognition module analyzes the preliminary global map, detects possible mid level objects and models ideal hypothetical sources. These can be seen as suggestions, fed back into the low level system by Virtual Scans. The low level system in turn adjusts its processing for re-evaluation by the mid level systems.

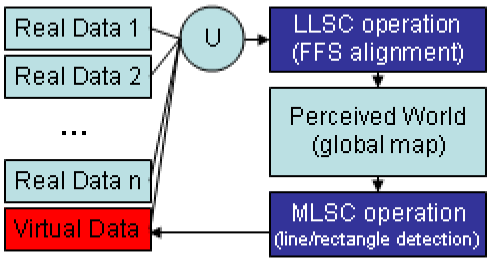

Fig. 5 LLSC/MLSC feedback. The LLSC module works on the union of real scans and the Virtual Scan. The MLSC module in turn re-creates a new Virtual Scan based on the result of the LLSC module.

In such a system, MLSC processes steer LLSC processes introducing higher knowledge to enable spatial inferences the LLSC system is not able to draw by itself. However, the MLSC system also needs assistance of the LLSC for two reasons: MLSC systems concentrate on higher information which needs LLSC pre-processed data (e.g. a set of collinear points is passed to the MLSC as a single line segment). But also LLSC processes have to support the suggestions stated by the MLSC. Since MLSC introduces higher knowledge, it is dangerous to focus on spatial mid level inferences too early. Feedback with the LLSC system enables more careful evaluation of plausibility.

The potential of MLSC has been largely unexplored in robotics, since recent research mainly addressed LLSC systems. They show an astonishing performance. But although the work on sophisticated statistical and geometrical models like extended Kalman Filters (EKF),e.g. [19], Particle Filters [15] and ICP (Iterative Closest Point) [7, 33] utilized in mapping approaches show impressive results, their limits are clearly visible. However, having these well-engineered low level systems at hand, it is natural to connect them to MLSC processes to mutually assist each other. In [6], the importance of 'Mental Imagery' in (Spatial) Cognition is emphasized and basic requirements of modeling are stated. Mental Images invent or recreate experiences resemble actually perceived events or objects. This is closely related to Virtual Scans.

4.1 Scan Alignment using Force Field Simulation

We assume the scans to be roughly pre-aligned. FFS alignment, in detail described in [26], is able to iteratively refine such an alignment based on the scan data only. In FFS, each single scan is seen as a non-deformable entity, a 'rigid body'. In each iteration, a translation and rotation is computed for each single scan simultaneously. This process minimizes a target function, the 'point potential', which is defined on the set of all data points (real and Virtual Scans: FFS can *not* distinguish). FFS solves the alignment problem as optimization problem utilizing a gradient descent approach motivated by simulation of dynamics of rigid bodies (the scans) in gravitational fields, but "*replaces laws of physics with constraints derived from human perception*" [26]. The gravitational field is based on a correspondence function between all pairs of data points, the 'force' function. FFS minimizes the overlaying potential function induced by the force and converges towards a local minimum of the potential, representing a locally optimal transformation of scans. The force function is designed in a manner that a low potential corresponds to a visually good appearance of the global map. As scans are moved according to the laws of motion of rigid bodies in a force field, single scans are not deformed.

4.2 Augmenting Data using Virtual Scans

The analysis module detects line segments and rectangles in each iteration of the FFS alignment. Both detection steps work on the entire point set of the current global map, i.e. the union of all points of the real scans. A preprocessing step transforms the point-based data to line segments. Similar segments are merged. This simplified data set allows for fast detection of lines and rectangles using techniques based on [17] and [25] respectively.

A Virtual Scan is a set of virtual laser scan points, superimposed over the entire area of the global map. The detected line segments and rectangles are 'plotted' into the Virtual Scans, i.e. they are represented by point sets as if they would be detected by a laser scanner.

An important feature of a Virtual Scan is, that each point in the Virtual Scan is assigned a weight, being the strength of hypothesis of the virtual structure it represents. Figure 6 shows an example. This data set consists of 60 single laser scans. The scans resemble the situation of an indoors disaster scenery, scanned by multiple robots. We used an initial rough guess of robot poses as global map for two different runs of FFS, once with Virtual Scans, once without. The experiment was performed to demonstrate the increase in alignment performance using Virtual Scans. The in-

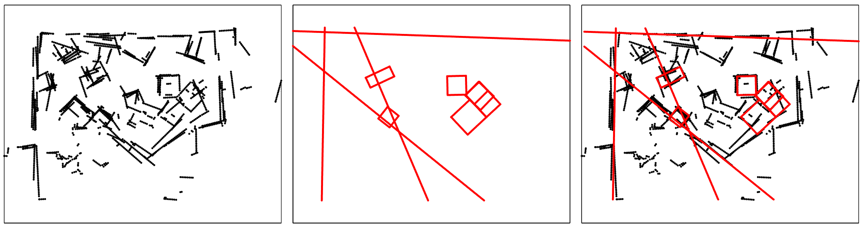

Fig. 6 Virtual Scans in an early stage of FFS. a) global map b) the Virtual Scan consisting of points representing detected lines and rectangles c) superimposition of real data and Virtual Scan. This is the data used in the next FFS iteration.

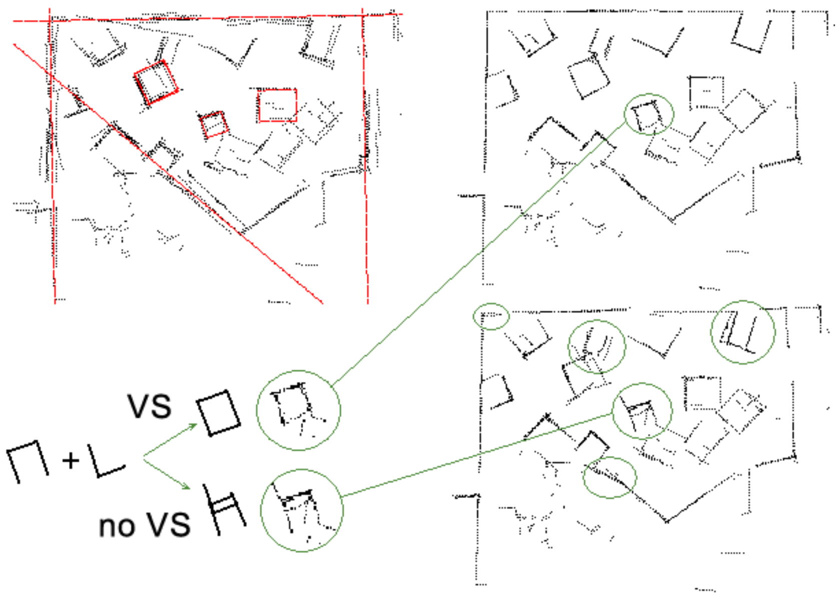

Fig. 7 Alignment of NIST data set (initial alignment see fig. 6). Top left: after 10 iterations with detected line and rectangular objects plotted into the Virtual Scan (red). Top right: Final result using Virtual Scans, after 100 iterations. MLSC objects are not shown for clarity of display. Compare to Bottom Right: final result of alignment without Virtual Scans. Encircled areas show examples of improvement using Virtual Scans. Bottom left: The center rectangle could only be aligned correctly using MLSC information.

crease in performance was evaluated by visual inspection, since for this data set no ground truth data is available. Comparing the final global maps of both runs, the utilization of Virtual Scans leads to distinct improvement in overall appearance and mapping details, see fig.7. Overall, the map is more 'straight' (compare e.g. the top wall), since the detection of globally present linear structures (top and left wall in fig.7) adjusts all participating single segments to be collinear. These corrections advance into the entire structure. More objectively, the improvements can be seen in certain details, the most distinct encircled in fig.7, bottom right. There especially the rectangle in the center of the global map is an excellent example for a situation where correct alignment is not achievable with low level knowledge only. Only the suggested rectangle from the Virtual Scan (see fig.7, top left) can force the low level process to transform the scan correctly. Without the assumed rectangle the low level optimization process necessarily tried to superimpose 2 parallel sides of the rectangle to falsely appear as one (fig.7,bottom right).

Comparison of fig.7, top left, and fig.6 shows the effect of feedback between the core FFS alignment process and the map analysis to create Virtual Scans. Figure 6 shows iteration 5 of the same experiment. Objects and object locations differ between the 2 Virtual Scans. Fig. 7 has discarded some hypotheses (objects) present in fig. 6, e.g. some of the rectangles. Other hypotheses are modified, e.g. the top wall is adjusted.

4.3 Map Evaluation using Virtual Scans

The idea of Virtual Scans can be used to evaluate the quality of mapping results in a straightforward way. This evaluation assumes the presence of a ground truth map G. To evaluate a mapping result R, it is fed into the FFS/VS system. G is used as Virtual Scan. Therefore, instead of *creating* a Virtual Scan, the ground truth data G is *inserted*, see fig.8 left. Assigning a high confidence weight to G will force the evaluated data R to align to the ground truth Virtual Scan. The alignment energy for this process is directly readable from the FFS module. The energy is a measure for visual closeness, see fig.8 right.

This evaluation procedure is adjustable to local ground truth data, since the adjustment energy in regions of interest can be emphasized. The energy computed in FFS is a symmetric measure, i.e. aligning R to G leads to the same measure as aligning G to R. This can be used for 'inverse evaluation' (evaluating the ground truth G with R) in the following manner: G can be manually split into local regions of interest (room, hallway, etc.). These regions are represented as single scans and used as input for the FFS system, while the map R is used as Virtual Scan. Such a setting has huge advantages, since, on one hand, it is more independent of the actual data representation of R. On the other hand, and more important, the manual split of G defines regions which can be assigned independent evaluation scores.

Note that in the Virtual Scan evaluation approach, ground truth data representation is not limited to *physical objects*, but can consist of *geometric properties* (e.g. evaluate how well map represents lines/rectangles). In this case, the original Virtual Scan approach is utilized, instead of insertion. The properties are then defined by means of MLSC-analysis modules. This is especially interesting if no ground truth data is available.

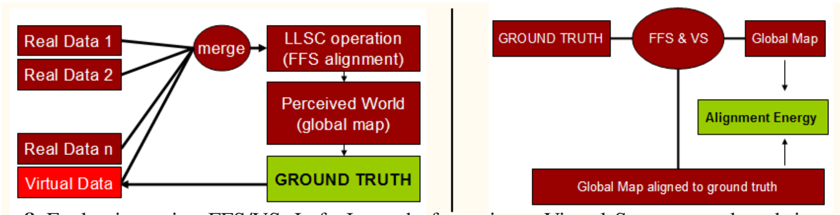

Fig. 8 Evaluation using FFS/VS. Left: Instead of creating a Virtual Scan, ground truth is used. Right: the alignment energy gained from FFS is a measure for visual fitness of the evaluated global map.

5 Evaluating Global Metric Consistency of Robot-Generated Maps

Fig. 9 Ground truth map.

Assessing the global consistency of robot generated maps is one of the practical ways to assess the capability of a robot's understanding of its surrounding environment. Global map quality is one of the quantitative measures which can be helpful in determining which robots will perform better in the field. One important factor which we want to measure in the global map quality for robot is the structural details in the map, so although there might be some other noise in the map it is assumed that any map is accurate if it thoroughly represents all the important structure features when compared to the ground truth map (Figure 5). We are proposing a novel method to assess the map quality based on three separate algorithms, each corresponding to different type of features found in the map. These are Harris Corner

Detector, Hough Transforms and Scale Invariant Feature transform. These measures will gives us three values which can be used to assess the quality of the map.

5.1 Harris-based Algorithm

Our first algorithm is defined on the principals of closest point matching. Let us assume we are given two images to compare named X and Y. To compare these images we need to find interest points in these images. These images are binary so we have limited choice in selecting the interest point algorithms. Most of the interest point detectors work on gray scale or color images. The interest points should be useful with enough detail so that they can be compared with points in other image.

Corner detectors are effective in case we have binary images so we have chosen Harris corner detector [16, 39]. This algorithm is very effective in capturing corners and is effectively invariant to rotation, scale, illumination variation and image noise. This is a desirable metric which will enable us to deal with minor noise, rotation and scale problems in the map.

After calculating the interest point using the Harris corner detector, we use the closest point matching process (described in Section 5.4 to generate the vector maps which are later used for calculating the quality metric (described in Section **??**. To generate the vector map (described in Sectio 5.5 we find the corners which are closest to the point under consideration and then use that point in map and find its closest point in the ground truth and eliminate those points from both maps with increase in the value for true points matched counter for the map quality.

5.2 Hough-based Algorithm

To account for the structural detail we have used Hough transform [22, 5] to transfer the map from Euclidean space to Hough space. This has the benefit of identifying lines in the image. These lines are compared according to the position of lines as points in the Hough space. Hough space is created by exchanging the Euclidean coordinates with the parameterized values form the parametric form of the equation of the line.

$$r_\theta = x \times \cos\theta + y \times \sin\theta \tag{6}$$

This helps in identifying lines easily as in the Hough space the points with large values will be highly likely to represent the lines.

This same process can be repeated to generate the space for circle and other geometrical objects detection. A variation of the Hough transform which is known as the generalized Hough transform, can be used to detect different type of arbitrary shapes in the image. This can be used to detect lines, squares (e.g. rooms), circle (e.g. roundabouts) in the map which will be a more generalized way to calculate the map quality.

After detection of these features the matching features can be located in the ground truth map and compared for the map quality as described in the last section.

5.3 Scale Invariant Feature Transform

$$DOG(I) = (G_m \times I) - (G_n \times I) \tag{7}$$

Scale Invariant Feature Transform (SIFT) was introduced by the David Lowe in [28]. Since then SIFT based localized feature have gained prominence among researchers due to there invariability to rotation scale and even dynamic changes. To assess the map quality we have proposed an algorithm based the SIFT. SIFT feature are calculated from extrema detection by finding the extrema points from difference of Gaussian images as shown in Equation 7, where the *Gm* and *Gn* represent the Gaussian filters at multiple scales and *I* is the original image. These points are further processed to find out the stable point under various conditions like edge response and low contrast point elimination.

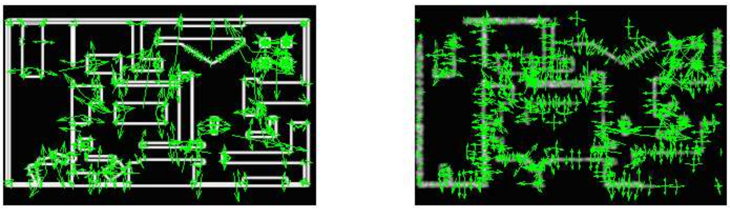

Fig. 10 Scale Invariant Feature transform is used to extract features. The figure on the left shows the results of SIFT on the ground truth map and the figure on the right shows the results of SIFT on a robot-generated map.

SIFT points detection is the first part of the process, after detection usually a descriptor is calculated and stored for each point so that it can be used to compare point from different images. The length of the SIFT detector is equal to 128 elements, which is basically the directional histogram of the local region.

For our algorithm we have used the following procedure:

1. First the entropy [14] of the image is calculated so that important regions with high entropy are identified. As our maps are binary images it is necessary to convert them into multiple scales with more information so that useful features are calculated.

2. This image is passed on to the SIFT for feature detection and descriptor calculation, see Figure 10 for an example of SIFT features.

5.4 Closest Point Matching

Closest point matching is performed by finding the closest point to the corresponding interest points in one image to another. Each point in the ground truth is mapped in a one to one fashion between the ground truth image and the target image. To keep points from matching to a point which is extremely far, the matching is performed only for the points which exist below a specified threshold. So it generates a displacement map for each point from one image to another image. The obvious benefit is the localized identification of the object interest points.

The closest point match can be described by Equation 8.

$$Match = Dis(FV(P(x,y))) - FV(P_\theta(x,y))) \tag{8}$$

Where Equation 8 describes that the match is the point which is equivalent to the point in one map to the corresponding region in another map under an specified threshold, where FV is the feature vector of the $P(x,y)$ and Dis is the distance between two corresponding feature vectors. Only in the case of the SIFT features the comparing criteria is based on the calculated descriptors

5.5 Vectorial Space

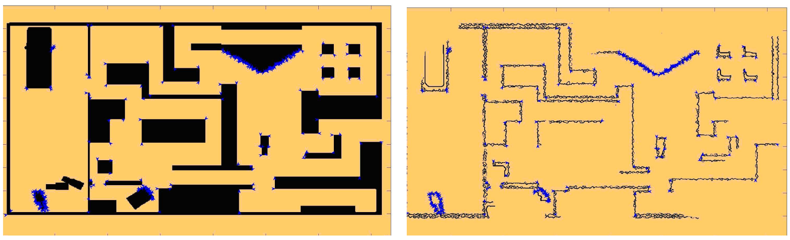

Fig. 11 The figure on the left shows the displacment of closest points in ground truth map. The figure on the right shows the displacment of the points in robot-generated map. points of interest indentified on the robot-generated map.

The displacement or vector map calculated in the last step provides much more information regarding the kind of distortion which appeared in the image. This way this vector map is a localized distortion map in the image. This can be done in both directions to identify the missing features which were not captured and extra features which don't really exist. The Figure 11 shows the displacement of closest points in ground truth while the vectorial space is shown in Figure 11 for the test image.

5.6 Quality Measure

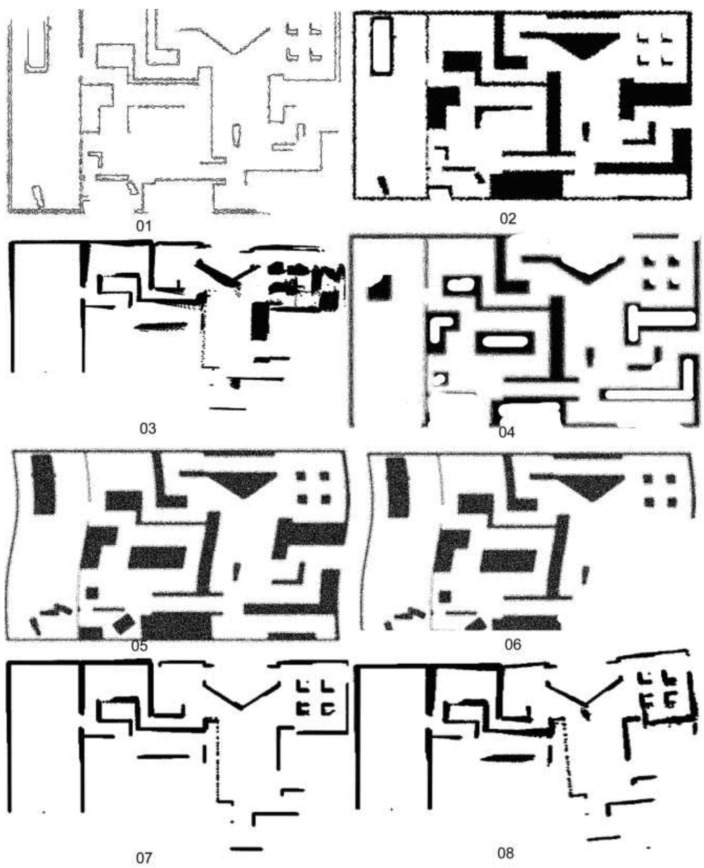

Fig. 12 Maps used for the comparison.

The map quality measure is calculated using the ratio between the set of features. The map quality can be defined mathematically as

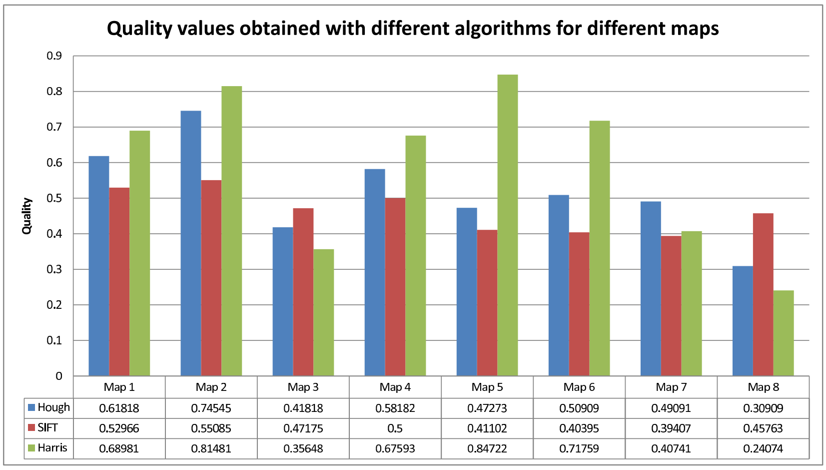

Fig. 13 Quality values obtained with different algorithms for different maps.

$$q = \frac{RMF}{GTF} \tag{9}$$

where RMF are the number of valid feature points found in the robot generated map while GTF are the number of feature points in the ground truth map. The map quality obtained from the set of test images (as show in Figure 12) is shown Figure 13.

Image quality assessment is difficult [42, 43] because for each case there can be different criteria to define the quality. For these robot generated maps the most important quality measure is the amount of features or landmarks (e.g. points, lines, etc) which are contained in the generated map. That is why we have based our quality measure on the feature having same shapes. We have not used the texture and color information because the maps are only binary images.

A very subtle issue is with the finding of the quality of the maps when they are the subset of a larger map. The ground truth is assumed to be the superset of all the maps so it contains all the features and information. So to assess the quality of the map which is smaller than the ground truth, we have to identify the subset from ground truth for which the map was generated. This remains an issue with this algorithm although for maps which are equivalent to the ground truth the algorithm gives fairly accurate results.

Only other remaining issue is the utilization of the threshold. Utilization of threshold can be a problem because we will not be able to match features if the maps are not aligned as in the case of Harris and Hough transform but this is not the case for SIFT based detector because it can detect matches even if they are far away, independent of scale, rotation and dislocation. Although for the Harris and Hough alignment of the map remains an important point. Alignment can be achieved by a startup marker that identifies a stable point between the robot generated map and the ground truth. A map can be considered more accurate if it consistently shows good performance in all three measures.

5.7 Limitations

This system is only suitable for offline measurement for the quality of the maps. As per definition the measure of quality is very difficult to define because requirements on which the map quality is based can be changed according to the need.

This algorithm measures the quality only on the basis of the information content of the image. These maps only contain bi-level images without any additional information. Map distortions and noise are not considered because the information is intact even with the added noise.

Some of the limitations which are observed are due to the type of maps used for processing. If the map has noise, such as, a jagged line or map with distortions, most likely the Harris corner detector will find lots of corners which could give erroneous results. Also Hough transform will fail for the case when point cloud data is separated quit far apart. Similarly for the SIFT case, if there is too much noise in the maps, this will introduce additional features which can cause problems during comparison of the features, because closely related features will give similar results.

6 Conclusion

Motivated by the absence of how one would evaluate robot-generated maps in a quantitative sense, our efforts have focused on bringing together the research community to collectively address this problem. This Chapter discussed our recent efforts in addressing this problem based on a recursive developmental cycle that encompasses standard test methods and objective evaluation methodologies.

Based on our previous work on performance singularity identification and testing, three test scenarios were developed by accounting for environmental conditions that robots typically encounter in unstructured domains. These scenarios considered the cases where there were distinct features readily available for reliable correspondence determination, occluded features that provided a considerable degree of difficulty, and a pathological case where establishing correspondences was extremely difficult. By varying the degree of difficulty, one is able to evaluate and analyze the robustness of mapping approaches.

Theoretical measures were developed using the Cramer-Rao Bound to arrive at a lower bound accuracy to compare experimental results. It was shown how the CRB bounds can be useful for assessing localization, pose tracking, and mapping estimates. Force Field Simulation was proposed as a methodology for assessing consistency of maps. By using the concept of Virtual Scans within the iterative FFS algorithm and experimental data, it was demonstrated how to evaluate the consistency of map quality at a local (metric) level. Three measures were then proposed to quantitatively asses global (metric) map quality using features extracted by three different detectors namely, the Harris corner detector, the hough transform and the scale invariant feature transform with respect to a ground truth map.

Our continuing research efforts will focus on how we can better refine the test methods and improve the map evaluation process. By working closely with the research community to develop standardized tools via standard test methods and reference data sets, researchers and consumers alike will be able to better evaluate the cost and benefits associated with robotic mapping technologies based on the end-user constraints.

References

1. Jacobs Map Analysis Toolkit. `http://usarsim.sourceforge.net/`
2. OpenSLAM. `http://www.openslam.org/`
3. Radish: The Robotics Data Set Repository. `http://radish.sourceforge.net/`
4. In: Form and style for ASTM Standards. ASTM International (2007)
5. Ballard, D.H.: Generalizing the Hough Transform to Detect Arbitrary Shapes pp. 714–725 (1987)
6. Bertel, S.: Thomas barkowsky, dominik engel, christian freksa. computational modeling of reasoning with mental images: basic requirements. D. Fum, F. del Missier, A. Stocco (Eds.), Proceedings of the 7th International Conference on Cognitive Modeling ICCM06 (2006)
7. Besl, P.: N. mckay. a method for registration of 3.d shapes. IEEE PAMI, 14(2) (1992)
8. C. Scrapper, R. Madhavan and S. Balakirsky (Organizers): Special Session on ‘Quantitative Assessment of Robot-generated Maps’. Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop, R. Madhavan and E. Messina (eds.); `http://www.isd.mel.nist.gov/PerMIS\_2008/` (2008)
9. Censi, A.: On achievable accuracy for range-finder localization. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 4170–4175. Rome, Italy (2007). DOI 10.1109/ROBOT.2007.364120. URL `http://purl.org/censi/2006/accuracy`. See also expanded version at http://purl.org/censi/2007/accuracy
10. Censi, A.: On achievable accuracy for pose-tracking. In: Proceedings of the IEEE International Conference on Robotics & Automation (ICRA) (2009). URL `http://purl.org/censi/2006/icpcov`
11. Crisan, D., Doucet, A.: A survey of convergence results on particle filtering for practitioners. IEEE Transactions on Signal Processing **50**(3), 736–746 (2002). URL `citeseer.ist.psu.edu/crisan02survey.html`
12. Fox, D.: Adapting the sample size in particle filters through KLD-sampling. International Journal of Robotics Research **22**(12) (2003)
13. Frese, U.: A discussion of simultaneous localization and mapping. Autonomous Robots **20**(1), 25–42 (2006)
14. Gonzalez, R., Woods, R., Eddins, S.: Chapter 11 pp. 714–725 (2003)
15. Grisetti, G.: C. stachniss, burgard w. improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. ICRA (2005)
16. Harris, C., Stephens, M.: A combined corner and edge detection. In: Proceedings of The Fourth Alvey Vision Conference, pp. 147–151 (1988)
17. Hough, P.V.C.: Methods and means for recognizing complex patterns. US patent 3,069,654 (1962)

18. Huang, G., Mourikis, A., Roumeliotis, S.: Analysis and improvement of the consistency of Extended Kalman Filter based SLAM. In: Proceedings of the IEEE International Conference on Robotics & Automation (ICRA), pp. 473–479 (2008). DOI 10.1109/ROBOT.2008.4543252
19. Huang, S.: G. dissanayake. convergence analysis for extended kalman filter based slam. IEEE International Conference on Robotics and Automation (2006)
20. Huang, S., Dissanayake, G.: Convergence and consistency analysis for Extended Kalman Filter based SLAM. IEEE Transactions on Robotics **23**(5), 1036–1049 (2007). DOI 10.1109/TRO.2007.903811
21. I. Varsadan, A. Birk, and M. Pfingsthorn: Determining Map Quality through an Image Similarity Metric. In: Proceedings of the RoboCup Symposium (2008)
22. Illingworth, J., Kittler, J.: A Survey of the Hough Transform. Computer Vision, Graphics, and Image Processing **44**(1), 87–116 (1988)
23. J. Weingarten and R. Siegwart: EKF-based 3D SLAM for Structured Environment Reconstruction. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3834–3839 (2005)
24. Joshi, S., Kaziska, D., Srivastava, A., Mio, W.: Riemannian structures on shape spaces: A framework for statistical inferences. Statistics and Analysis of Shapes pp. 313–333 (2006)
25. Lagunovsky, D.: Ablameyko s. fast line and rectangle detection by clustering and grouping. Proc. of CAIP'97, Kiel, Germany (1997)
26. Lakaemper, R.: N.adluru, l.j.latecki, r.madhavan. multi robot mapping using force field simulation. Journal of Field Robotics, Special Issue on Quantitative Performance Evaluation of Robotic and Intelligent Systems (2007)
27. Le, H., Kendall, D.G.: The Riemannian structure of Euclidean shape spaces: A novel environment for statistics. Annals of Statistics **21**(3), 1225–1271 (1993)
28. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
29. M. Chandran-Ramesh and P. Newman: Assessing Map Quality using Conditional Random Fields. In: Proceedings of the Field and Service Robotics (2007)
30. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. Journal of the European Mathematical Society **8**, 1–48 (2003)
31. Mourikis, A.: Personal communication (2008)
32. Mourikis, A., Roumeliotis, S.: On the treatment of relative-pose measurements for mobile robot localization. In: Proceedings of the IEEE International Conference on Robotics & Automation (ICRA). Orlando, FL (2006)
33. Nuechter, A.: K. lingemann, j. hertzberg, h. surmann, k. pervoelz, m. hennig, k. r. tiruchinapalli, r. worst, t. christaller. mapping of rescue environments with kurt3d. Proceedings of the International Workshop on Safety, Security and Rescue Robotics (SSRR '05),Kobe, Japan (2005)
34. R. Madhavan, C. Scrapper and A. Kleiner (eds.): Quantitative Performance Evaluation of Navigation Solutions for Mobile Robots. Workshop Proceedings, Robotics: Science and Systems (RSS) Conference, Zurich, Switzerland; `http://kaspar.informatik.uni-freiburg.de/~rss/` (2008)
35. S. Abdallah, D. Asmar, and J. Zelek: Towards Benchmarks for Vision SLAM Algorithms. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 3834–3839 (2006)
36. S. Thrun: Learning Metric-Topological Maps for Indoor Mobile Robot Navigation. Artificial Intelligence pp. 21–71 (1998)
37. Scrapper, C., Madhavan, R., Balakirsky, S.: Stable Navigation Solutions for Robots in Complex Environments. In: IEEE International Workshop on Safety, Security, and Rescue Robotics (SSRR2007) (2007)
38. Scrapper, C., Madhavan, R., Balakirsky, S.: Performance analysis for stable mobile robot navigation solutions. p. 696206. SPIE (2008)
39. Trajkovic, M., Hedley, M.: Fast corner detection". Image and Vision Computing **16**, 75–87 (1998)

40. Trees, H.L.V., Bell, K.L.: Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking. Wiley-IEEE Press (2007)
41. W. Zhou, J. Miro, and G. Dissanayake: Information Efficient 3D Visual SLAM in Unstructured Domains. In: Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information, pp. 323–328 (2007)
42. Wang, Z., Bovik, A.: A Universal Image Quality Index. In: IEEE Signal Processing Letters, vol. 9, pp. 81–84 (2002)
43. Wang, Z., Bovik, A.C.: Why is Image Quality Assessment so Difficult. In: in Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing, pp. 3313–3316 (2002)